

# CC3301

## Programación de Software de Sistemas

### Profesor: Luis Mateu

- El preprocesador de C
- Macros con parámetros
- Compilación condicional
- Compilación integrada vs. separada

# El preprocesador de C

- Es la primera etapa de la compilación
- Procesa directivas que empiezan con #
- Expande directivas como: `#include <stdio.h>`
- Reemplaza literalmente la directiva por el contenido del archivo `stdio.h`
- Definición de macros: `#define N 30`
- Cada vez que se encuentre `N` en el archivo se reemplaza literalmente por `30`

# Macros con parámetros

- Definición de macros con parámetros:

- `#define MAX(i,j) i>j ? i : j`

`int c=1+MAX(a,N);` → `int c=1+a>30 ? a : 30;`

- Para evitar problemas de parentización, prefiera esta definición:

- `#define MAX(i,j) ((i)>(j)) ? (i) : (j)`

`1+MAX(a,N)` → `1+((a)>(30)) ? (a) : (30)`

- Problema:

`1+MAX(a++,N);` → `1+((a++)>(30)) ? (a++) : (30);`

# Compilación condicional

<code>#ifdef N</code>	<code>#ifndef N</code>	<code>#if 1</code>
<code>char buf[N];</code>	<code># define N 100</code>	<code>printf("hola\n");</code>
<code>#else</code>	<code>#endif</code>	<code>#endif</code>
<code>char buf[100];</code>	<code>char buf[N];</code>	
<code>#endif</code>		<code>#if 0</code>
		<code>x= 1;</code>
		<code>#endif</code>

- Ver resultado del preproceso:

```
gcc -E prog.c | less
```

- Usualmente el archivo es enorme: presione la tecla G mayúscula para ir al final

# Compilación integrada vs. separada

- Compilación integrada:

```
$ gcc test-alinear.c alinear.c -lm -o alinear.bin
```

- Cada archivo se compila de manera independiente del resto de los archivos

- Compilación separada:

```
$ gcc -c test-alinear.c
```

```
$ gcc -c alinear.c
```

```
$ gcc test-alinear.o alinear.o -o alinear.bin
```

- La última fase se llama *link*
- Más información [acá](#)